

## HANDWRITTEN TEXT CAPTURE VIA INTERFACE SURFACE

### FIELD OF INVENTION

The present invention relates to a method and system for enabling user interaction with computer software running in a computer system.

- 5           The invention has been developed primarily to provide the basis for a surface-based interface which allows a user to interact with networked information and to obtain interactive printed matter on demand via high-speed networked color printers. Although the invention will largely be described herein with reference to this use, it will be appreciated that the invention is not limited to use in this field.

### 10 ~~CO-PENDING APPLICATIONS~~

Various methods, systems and apparatus relating to the present invention are disclosed in the following co-pending applications filed by the applicant or assignee of the present invention simultaneously with the present application:

- 15           NPA001US, NPA002US, NPA003US, NPA004US, NPA005US, NPA006US, NPA007US, NPA008US, NPA009US, NPA010US, NPA012US, NPA016US, NPA017US, NPA018US, NPA019US, NPA020US, NPA021US, NPA030US, NPA035US, NPA048US, NPA050US, NPA051US, NPA052US, NPA075US, NPB001US, NPB002US, NPK002US, NPK003US, NPK004US, NPK005US, NPK007US, NPM001US, NPM002US, NPM003US, NPM004US, NPN001US, 20           NPN002US, NPN003US, NPP001US, NPP002US, NPP003US, NPP005US, NPP006US, NPP007US, NPP008US, NPP016US, NPP017US, NPP018US, NPP019US, NPS001US, NPS003US, NPS020US, NPT001US, NPT002US, NPT003US, NPT004US, NPX001US, NPX003US, NPX008US, NPX011US, NPX014US, NPX016US, NPX020US, NPX022US, IJ52US, IJM52US, 25           MJ10US, MJ11US, MJ12US, MJ13US, MJ14US, MJ15US, MJ34US, MJ47US, MJ52US, MJ58US, MJ62US, MJ63US, PAK04US, PAK05US, PAK06US, PAK07US, PAK08US, PEC01US, PEC02US.

- 30           The disclosures of these co-pending applications are incorporated herein by cross-reference. Each application is temporarily identified by its docket number. This will be replaced by the corresponding USSN when available.

## BACKGROUND

Presently, users wishing to input text into a computer system typically do so via a keyboard. Whilst such interface systems are useful, they are relatively bulky, and lack the tactile nature of using a pen to record information on paper. Moreover, in many situations, paper used in conjunction with a pen is a more physically comfortable and convenient information input tool, due to its visual characteristics and its portability compared to computer keyboards and other computer interface devices. However, paper-based recording of information lacks many of the advantages of computer-based recording, such as searchability and compact archiving. Unlike keyboard input systems, pen and paper generally lack the ability to control computer software.

## SUMMARY OF INVENTION

According to a first aspect, there is provided a method of enabling user interaction with computer software running in a computer system via:

an interface surface containing information relating to the computer software and including coded data indicative of a text field; and

a sensing device which, when placed in an operative position relative to the interface surface, senses indicating data indicative of the text field and generates movement data indicative of the sensing device's movement relative to the interface surface;

the method including the steps of, in the computer system:

- (a) receiving the indicating data from the sensing device;
- (b) receiving the movement data from the sensing device;
- (c) identifying the text field from the indicating data; and
- (d) operating the computer software at least partly in reliance on the movement data, and in accordance with instructions associated with the text field.

Preferably, the first aspect further includes the steps of, in the computer system, performing text recognition on the handwritten user input, thereby to generate computer text.

Preferably, the text field is associated with a visible text zone defined on the

interface surface.

In a preferred embodiment, the first aspect further includes the step of, in the computer system, recognising whether the movement data is indicative of an text editing command drawn onto the surface by the user. More preferably, in the event that an editing command is recognised, the computer system is operated in accordance with instructions associated with the editing command. Preferably, the editing command is selected from the following group: strikeout; underlining; cutting; pasting; and relocation.

According to a second aspect of the invention, there is provided a system for enabling user interaction with computer software running in a computer system via:

an interface surface containing information relating to the computer software and including coded data indicative of a text field; and

a sensing device which, when placed in an operative position relative to the interface surface, senses indicating data indicative of the text field and generates movement data indicative of the sensing device's movement relative to the interface surface;

the computer system being configured to:

- (a) receive the indicating data from the sensing device;
- (b) receive the movement data from the sensing device;
- (c) identify the text field from the indicating data; and
- (d) operate the computer software at least partly in reliance on the movement data, and in accordance with instructions associated with the text field.

According to a second aspect of the invention, there is provided a system for enabling user interaction with computer software running in a computer system, the system including:

an interface surface containing information relating to the computer software and including coded data indicative of a text field relating to the computer software;

the system being configured to, in the computer system:

- 5 (a) receive indicating data from a sensing device, the indicating data being indicative of the text field, wherein the sensing device, when placed in an operative position relative to the interface surface, senses the indicating data and generates movement data indicative of the sensing device's movement relative to the interface surface;
- (b) receive the movement data from the sensing device;
- (c) identify the text field on the basis of the indicating data; and
- (d) operate the computer software at least partly in reliance on the movement data, and in accordance with instructions associated with the text field.

10 Preferably, in the second and third aspects, the computer system is configured to perform text recognition on the handwritten user input, thereby to generate computer text.

In preferred forms of the second and third aspects, the computer system is configured to recognise whether the movement data is indicative of an editing command drawn onto the surface by the user. More preferably, in the event that an editing command is recognised, the computer software is operated in accordance with instructions associated with the editing command. Preferably, the editing command is selected from the following group in relation to text: strikeout; underlining; cutting; pasting; and relocation.

15

20 In preferred embodiments of the second and third aspects, the text field is associated with a visible text zone defined on the interface surface.

In a particularly preferred form, the coded data takes the form of tags printed onto a piece of paper, the tags being configured to be read by a sensing device in the form of an optical sensing stylus. The tags are preferably printed using an ink that absorbs near infrared light but is substantially invisible to a human viewer under normal lighting conditions. When a user brings a sensing end of the stylus close to the surface, one or more of the tags are imaged, interpreted and decoded to provide an indication of the text field.

25

The movement data can be generated in any of a number of ways. In one form, the coded data includes information indicative of positions of points within the region,

30



10

Figure 3 illustrates a collection of netpage servers and printers interconnected via a network;

Figure 5 is a plan view showing a structure of a netpage tag;

20 Figure 7 is a flowchart of a tag image processing and decoding algorithm;

Figure 9 is a perspective exploded view of the netpage pen shown in Figure 8;

Figure 11 is a perspective view of a wall-mounted netpage printer;

Figure 12a is an enlarged portion of Figure 12 showing a section of the duplexed print



5

Memjet™ printheads associated with the printer controller shown in Figure 14;

14 and 15;

10

elements;

15

Figure 20 is a perspective view of a short segment of a pagewidth Memjet™ printhead:

Figure 21 is a schematic view of a user class diagram;

Figure 22 is a schematic view of a printer class diagram;

Figure 23 is a schematic view of a pen class diagram;

20

Figure 25 is a schematic view of a document and page description class diagram;

Figure 26 is a schematic view of a document and page ownership class diagram;

Figure 27 is a schematic view of a terminal element specialization class diagram;

Figure 28 is a schematic view of a static element specialization class diagram;

25

Figure 30 is a schematic view of a hyperlink element specialization class diagram:

Figure 31 is a schematic view of a hyperlinked group class diagram;

Figure 33 is a schematic view of a digital ink class diagram;

Figure 35 is a schematic view of a checkbox field class diagram;

Figure 37 is a schematic view of a signature field class diagram;

Figure 38 is a flowchart of an input processing algorithm;

Figure 38a is a detailed flowchart of one step of the flowchart of Figure 38;

Figure 39 is a schematic view of a page server command element class diagram;

10 Figure 40 is a schematic view of a resource description class diagram;

Figure 41 is a schematic view of a favorites list class diagram;

Figure 42 is a schematic view of a history list class diagram;

Figure 43 is a schematic view of a subscription delivery protocol;

Figure 44 is a schematic view of a hyperlink request class diagram;

15 Figure 45 is a schematic view of a hyperlink activation protocol;

Figure 46 is a schematic view of a form submission protocol; and

Figure 47 is a schematic view of a commission payment protocol.

## DETAILED DESCRIPTION OF PREFERRED AND OTHER EMBODIMENTS

Note: Memjet™ is a trade mark of Silverbrook Research Pty Ltd, Australia.

In the preferred embodiment, the invention is configured to work with the netpage networked computer system, a detailed overview of which follows. It will be appreciated that not every implementation will necessarily embody all or even most of the specific details and extensions discussed below in relation to the basic system. However, the system is described in its most complete form to reduce the need for external reference when attempting to understand the context in which the preferred embodiments and aspects of the present invention operate.

10 In brief summary, the preferred form of the netpage system employs a computer interface in the form of a mapped surface, that is, a physical surface which contains references to a map of the surface maintained in a computer system. The map references can be queried by an appropriate sensing device. Depending upon the specific implementation, the map references may be encoded visibly or invisibly, and defined in  
15 such a way that a local query on the mapped surface yields an unambiguous map reference both within the map and among different maps. The computer system can contain information about features on the mapped surface, and such information can be retrieved based on map references supplied by a sensing device used with the mapped surface. The information thus retrieved can take the form of actions which are initiated  
20 by the computer system on behalf of the operator in response to the operator's interaction with the surface features.

In its preferred form, the netpage system relies on the production of, and human interaction with, netpages. These are pages of text, graphics and images printed on ordinary paper, but which work like interactive web pages. Information is encoded on  
25 each page using ink which is substantially invisible to the unaided human eye. The ink, however, and thereby the coded data, can be sensed by an optically imaging pen and transmitted to the netpage system.

In the preferred form, active buttons and hyperlinks on each page can be clicked with the pen to request information from the network or to signal preferences to a  
30 network server. In one embodiment, text written by hand on a netpage is automatically recognized and converted to computer text in the netpage system, allowing forms to be



As illustrated in Figure 1, a printed netpage 1 can represent a interactive form which can be filled in by the user both physically, on the printed page, and “electronically”, via communication between the pen and the netpage system. The example shows a “Request” form containing name and address fields and a submit button. The netpage consists of graphic data 2 printed using visible ink, and coded data 3 printed as a collection of tags 4 using invisible ink. The corresponding page description 5, stored on the netpage network, describes the individual elements of the netpage. In particular it describes the type and spatial extent (zone) of each interactive element (i.e. text field or button in the example), to allow the netpage system to correctly interpret input via the netpage. The submit button 6, for example, has a zone 7 which corresponds to the spatial extent of the corresponding graphic 8.

The netpage printer 601, a preferred form of which is shown in Figures 11 to 13 and described in more detail below, is able to deliver, periodically or on demand, personalized newspapers, magazines, catalogs, brochures and other publications, all printed at high quality as interactive netpages. Unlike a personal computer, the netpage printer is an appliance which can be, for example, wall-mounted adjacent to an area where the morning news is first consumed, such as in a user's kitchen, near a breakfast table, or near the household's point of departure for the day. It also comes in tabletop, desktop, portable and miniature versions.

As shown in Figure 2, the netpage pen 101 interacts with the coded data on a  
30 printed netpage 1 and communicates, via a short-range radio link 9, the interaction to a  
netpage printer. The printer 601 sends the interaction to the relevant netpage page server

10 for interpretation. In appropriate circumstances, the page server sends a corresponding message to application computer software running on a netpage application server 13. The application server may in turn send a response which is printed on the originating printer.

5           The netpage system is made considerably more convenient in the preferred embodiment by being used in conjunction with high-speed microelectromechanical system (MEMS) based inkjet (Memjet™) printers. In the preferred form of this technology, relatively high-speed and high-quality printing is made more affordable to consumers. In its preferred form, a netpage publication has the physical characteristics of  
10 a traditional newsmagazine, such as a set of letter-size glossy pages printed in full color on both sides, bound together for easy navigation and comfortable handling.

The netpage printer exploits the growing availability of broadband Internet access. Cable service is available to 95% of households in the United States, and cable modem service offering broadband Internet access is already available to 20% of these.

15 The netpage printer can also operate with slower connections, but with longer delivery times and lower image quality. Indeed, the netpage system can be enabled using existing consumer inkjet and laser printers, although the system will operate more slowly and will therefore be less acceptable from a consumer's point of view. In other embodiments, the netpage system is hosted on a private intranet. In still other embodiments, the netpage

20 system is hosted on a single computer or computer-enabled device, such as a printer.

Netpage publication servers 14 on the netpage network are configured to deliver print-quality publications to netpage printers. Periodical publications are delivered automatically to subscribing netpage printers via pointcasting and multicasting Internet protocols. Personalized publications are filtered and formatted according to individual user profiles.

A netpage printer can be configured to support any number of pens, and a pen can work with any number of netpage printers. In the preferred implementation, each netpage pen has a unique identifier. A household may have a collection of colored netpage pens, one assigned to each member of the family. This allows each user to maintain a distinct profile with respect to a netpage publication server or application server.

A netpage pen can also be registered with a netpage registration server 11 and linked to one or more payment card accounts. This allows e-commerce payments to be securely authorized using the netpage pen. The netpage registration server compares the signature captured by the netpage pen with a previously registered signature, allowing it to authenticate the user's identity to an e-commerce server. Other biometrics can also be used to verify identity. A version of the netpage pen includes fingerprint scanning, verified in a similar way by the netpage registration server.

Although a netpage printer may deliver periodicals such as the morning newspaper without user intervention, it can be configured never to deliver unsolicited junk mail. In its preferred form, it only delivers periodicals from subscribed or otherwise authorized sources. In this respect, the netpage printer is unlike a fax machine or e-mail account which is visible to any junk mailer who knows the telephone number or email address.

## 1 NETPAGE SYSTEM ARCHITECTURE

Each object model in the system is described using a Unified Modeling Language (UML) class diagram. A class diagram consists of a set of object classes connected by relationships, and two kinds of relationships are of interest here: associations and generalizations. An association represents some kind of relationship between objects, i.e. between instances of classes. A generalization relates actual classes, and can be understood in the following way: if a class is thought of as the set of all objects of that class, and class A is a generalization of class B, then B is simply a subset of A. The UML does not directly support second-order modelling - i.e. classes of classes.

Each class is drawn as a rectangle labelled with the name of the class. It contains a list of the attributes of the class, separated from the name by a horizontal line, and a list of the operations of the class, separated from the attribute list by a horizontal line. In the class diagrams which follow, however, operations are never modelled.

An association is drawn as a line joining two classes, optionally labelled at either end with the multiplicity of the association. The default multiplicity is one. An asterisk (\*) indicates a multiplicity of "many", i.e. zero or more. Each association is optionally labelled with its name, and is also optionally labelled at either end with the role of the corresponding class. An open diamond indicates an aggregation association

A generalization relationship (“is-a”) is drawn as a solid line joining two classes, with an arrow (in the form of an open triangle) at the generalization end.

When a class diagram is broken up into multiple diagrams, any class which is  
5 duplicated is shown with a dashed outline in all but the main diagram which defines it. It  
is shown with attributes only where it is defined.

## 1.1 NETPAGES

Netpages are the foundation on which a netpage network is built. They provide a paper-based user interface to published information and interactive services.

10           A netpage consists of a printed page (or other surface region) invisibly tagged with references to an online description of the page. The online page description is maintained persistently by a netpage page server. The page description describes the visible layout and content of the page, including text, graphics and images. It also describes the input elements on the page, including buttons, hyperlinks, and input fields.

15   A netpage allows markings made with a netpage pen on its surface to be simultaneously captured and processed by the netpage system.

Multiple netpages can share the same page description. However, to allow input through otherwise identical pages to be distinguished, each netpage is assigned a unique page identifier. This page ID has sufficient precision to distinguish between a very large number of netpages.

Each reference to the page description is encoded in a printed tag. The tag identifies the unique page on which it appears, and thereby indirectly identifies the page description. The tag also identifies its own position on the page. Characteristics of the tags are described in more detail below.

25        Tags are printed in infrared-absorptive ink on any substrate which is infrared-reflective, such as ordinary paper. Near-infrared wavelengths are invisible to the human eye but are easily sensed by a solid-state image sensor with an appropriate filter.

A tag is sensed by an area image sensor in the netpage pen, and the tag data is transmitted to the netpage system via the nearest netpage printer. The pen is wireless and communicates with the netpage printer via a short-range radio link. Tags are sufficiently

5       The netpage page server maintains a unique page instance for each printed netpage, allowing it to maintain a distinct set of user-supplied values for input fields in the page description for each printed netpage.

10

### 1.2.1 Tag Data Content

15

20

30

Each tag contains 120 bits of information, typically allocated as shown in Table 1. Assuming a maximum tag density of 64 per square inch, a 16-bit tag ID supports a region size of up to 1024 square inches. Larger regions can be mapped continuously without increasing the tag ID precision simply by using abutting regions and maps. The 100-bit region ID allows  $2^{100}$  ( $\sim 10^{30}$  or a million trillion trillion) different regions to be uniquely identified.

### 1.2.2 Tag Data Encoding

The 120 bits of tag data are redundantly encoded using a (15, 5) Reed-Solomon code. This yields 360 encoded bits consisting of 6 codewords of 15 4-bit symbols each. The (15, 5) code allows up to 5 symbol errors to be corrected per codeword, i.e. it is tolerant of a symbol error rate of up to 33% per codeword.

### 1.2.3 Physical Tag Structure

NPX011US

To achieve proper tag reproduction, the tag is rendered at a resolution of 256×256 dots. When printed at 1600 dots per inch this yields a tag with a diameter of about 4 mm. At this resolution the tag is designed to be surrounded by a “quiet area” of radius 16 dots. Since the quiet area is also contributed by adjacent tags, it only adds 16 dots to the effective diameter of the tag.

The tag includes six target structures. A detection ring 15 allows the sensing device to initially detect the tag. The ring is easy to detect because it is rotationally invariant and because a simple correction of its aspect ratio removes most of the effects of perspective distortion. An orientation axis 16 allows the sensing device to determine the approximate planar orientation of the tag due to the yaw of the sensor. The orientation axis is skewed to yield a unique orientation. Four perspective targets 17 allow the sensing device to infer an accurate two-dimensional perspective transform of the tag and hence an accurate three-dimensional position and orientation of the tag relative to the sensor.

All target structures are redundantly large to improve their immunity to noise.

The overall tag shape is circular. This supports, amongst other things, optimal tag packing on an irregular triangular grid. In combination with the circular detection ring, this makes a circular arrangement of data bits within the tag optimal. To maximise its size, each data bit is represented by a radial wedge in the form of an area bounded by two radial lines and two concentric circular arcs. Each wedge has a minimum dimension of 8 dots at 1600 dpi and is designed so that its base (its inner arc), is at least equal to this minimum dimension. The height of the wedge in the radial direction is always equal to the minimum dimension. Each 4-bit data symbol is represented by an array of 2×2 wedges.

The 15 4-bit data symbols of each of the six codewords are allocated to the four concentric symbol rings 18a to 18d in interleaved fashion. Symbols are allocated alternately in circular progression around the tag.

The interleaving is designed to maximise the average spatial distance between any two symbols of the same codeword.

In order to support “single-click” interaction with a tagged region via a sensing

device, the sensing device must be able to see at least one entire tag in its field of view no matter where in the region or at what orientation it is positioned. The required diameter of the field of view of the sensing device is therefore a function of the size and spacing of the tags.

- 5            Assuming a circular tag shape, the minimum diameter of the sensor field of view is obtained when the tags are tiled on a equilateral triangular grid, as shown in Figure 6.

#### 1.2.4    Tag Image Processing and Decoding

- 10           The tag image processing and decoding performed by a sensing device such as the netpage pen is shown in Figure 7. While a captured image is being acquired from the image sensor, the dynamic range of the image is determined (at 20). The center of the range is then chosen as the binary threshold for the image 21. The image is then thresholded and segmented into connected pixel regions (i.e. shapes 23) (at 22). Shapes which are too small to represent tag target structures are discarded. The size and centroid  
15           of each shape is also computed.

Binary shape moments 25 are then computed (at 24) for each shape, and these provide the basis for subsequently locating target structures. Central shape moments are by their nature invariant of position, and can be easily made invariant of scale, aspect ratio and rotation.

- 20           The ring target structure 15 is the first to be located (at 26). A ring has the advantage of being very well behaved when perspective-distorted. Matching proceeds by aspect-normalizing and rotation-normalizing each shape's moments. Once its second-order moments are normalized the ring is easy to recognize even if the perspective distortion was significant. The ring's original aspect and rotation 27 together provide a  
25           useful approximation of the perspective transform.

- The axis target structure 16 is the next to be located (at 28). Matching proceeds by applying the ring's normalizations to each shape's moments, and rotation-normalizing the resulting moments. Once its second-order moments are normalized the axis target is easily recognized. Note that one third order moment is required to disambiguate the two  
30           possible orientations of the axis. The shape is deliberately skewed to one side to make



this possible. Note also that it is only possible to rotation-normalize the axis target after it has had the ring’s normalizations applied, since the perspective distortion can hide the axis target’s axis. The axis target’s original rotation provides a useful approximation of the tag’s rotation due to pen yaw 29.

The four perspective target structures 17 are the last to be located (at 30). Good estimates of their positions are computed based on their known spatial relationships to the ring and axis targets, the aspect and rotation of the ring, and the rotation of the axis. Matching proceeds by applying the ring's normalizations to each shape's moments. Once their second-order moments are normalized the circular perspective targets are easy to recognize, and the target closest to each estimated position is taken as a match. The original centroids of the four perspective targets are then taken to be the perspective-distorted corners 31 of a square of known size in tag space, and an eight-degree-of-freedom perspective transform 33 is inferred (at 32) based on solving the well-understood equations relating the four tag-space and image-space point pairs (see Heckbert, P., Fundamentals of Texture Mapping and Image Warping, Masters Thesis, Dept. of EECS, U. of California at Berkeley, Technical Report No. UCB/CSD 89/516, June 1989, the contents of which are herein incorporated by cross-reference).

The inferred tag-space to image-space perspective transform is used to project (at 36) each known data bit position in tag space into image space where the real-valued position is used to bilinearly interpolate (at 36) the four relevant adjacent pixels in the input image. The previously computed image threshold 21 is used to threshold the result to produce the final bit value 37.

Once all 360 data bits 37 have been obtained in this way, each of the six 60-bit Reed-Solomon codewords is decoded (at 38) to yield 20 decoded bits 39, or 120 decoded bits in total. Note that the codeword symbols are sampled in codeword order, so that codewords are implicitly de-interleaved during the sampling process.

The ring target 15 is only sought in a subarea of the image whose relationship to the image guarantees that the ring, if found, is part of a complete tag. If a complete tag is not found and successfully decoded, then no pen position is recorded for the current frame. Given adequate processing power and ideally a non-minimal field of view 193, an alternative strategy involves seeking another tag in the current image.

The obtained tag data indicates the identity of the region containing the tag and the position of the tag within the region. An accurate position 35 of the pen nib in the region, as well as the overall orientation 35 of the pen, is then inferred (at 34) from the perspective transform 33 observed on the tag and the known spatial relationship between the pen's physical axis and the pen's optical axis.

### 1.2.5 Tag Map

Decoding a tag results in a region ID, a tag ID, and a tag-relative pen transform. Before the tag ID and the tag-relative pen location can be translated into an absolute location within the tagged region, the location of the tag within the region must be known. This is given by a tag map, a function which maps each tag ID in a tagged region to a corresponding location. The tag map class diagram is shown in Figure 22, as part of the netpage printer class diagram.

A tag map reflects the scheme used to tile the surface region with tags, and this can vary according to surface type. When multiple tagged regions share the same tiling scheme and the same tag numbering scheme, they can also share the same tag map.

The tag map for a region must be retrievable via the region ID. Thus, given a region ID, a tag ID and a pen transform, the tag map can be retrieved, the tag ID can be translated into an absolute tag location within the region, and the tag-relative pen location can be added to the tag location to yield an absolute pen location within the region.

### 1.2.6 Tagging Schemes

Two distinct surface coding schemes are of interest, both of which use the tag structure described earlier in this section. The preferred coding scheme uses "location-indicating" tags as already discussed. An alternative coding scheme uses object-indicating tags.

A location-indicating tag contains a tag ID which, when translated through the tag map associated with the tagged region, yields a unique tag location within the region. The tag-relative location of the pen is added to this tag location to yield the location of the pen within the region. This in turn is used to determine the location of the pen relative to a user interface element in the page description associated with the region.

Not only is the user interface element itself identified, but a location relative to the user interface element is identified. Location-indicating tags therefore trivially support the capture of an absolute pen path in the zone of a particular user interface element.

5 An object-indicating tag contains a tag ID which directly identifies a user interface element in the page description associated with the region. All the tags in the zone of the user interface element identify the user interface element, making them all identical and therefore indistinguishable. Object-indicating tags do not, therefore, support the capture of an absolute pen path. They do, however, support the capture of a relative pen path. So long as the position sampling frequency exceeds twice the  
10 encountered tag frequency, the displacement from one sampled pen position to the next within a stroke can be unambiguously determined.

With either tagging scheme, the tags function in cooperation with associated visual elements on the netpage as user interactive elements in that a user can interact with the printed page using an appropriate sensing device in order for tag data to be read  
15 by the sensing device and for an appropriate response to be generated in the netpage system.

### 1.3 DOCUMENT AND PAGE DESCRIPTIONS

A preferred embodiment of a document and page description class diagram is shown in Figures 25 and 26.

20 In the netpage system a document is described at three levels. At the most abstract level the document 836 has a hierarchical structure whose terminal elements 839 are associated with content objects 840 such as text objects, text style objects, image objects, etc. Once the document is printed on a printer with a particular page size and according to a particular user's scale factor preference, the document is paginated and  
25 otherwise formatted. Formatted terminal elements 835 will in some cases be associated with content objects which are different from those associated with their corresponding terminal elements, particularly where the content objects are style-related. Each printed instance of a document and page is also described separately, to allow input captured through a particular page instance 830 to be recorded separately from input captured  
30 through other instances of the same page description.

The presence of the most abstract document description on the page server allows a user to request a copy of a document without being forced to accept the source document's specific format. The user may be requesting a copy through a printer with a different page size, for example. Conversely, the presence of the formatted document description on the page server allows the page server to efficiently interpret user actions on a particular printed page.

A formatted document 834 consists of a set of formatted page descriptions 5, each of which consists of a set of formatted terminal elements 835. Each formatted element has a spatial extent or zone 58 on the page. This defines the active area of input elements such as hyperlinks and input fields.

A document instance 831 corresponds to a formatted document 834. It consists of a set of page instances 830, each of which corresponds to a page description 5 of the formatted document. Each page instance 830 describes a single unique printed netpage 1, and records the page ID 50 of the netpage. A page instance is not part of a document instance if it represents a copy of a page requested in isolation.

A page instance consists of a set of terminal element instances 832. An element instance only exists if it records instance-specific information. Thus, a hyperlink instance exists for a hyperlink element because it records a transaction ID 55 which is specific to the page instance, and a field instance exists for a field element because it records input specific to the page instance. An element instance does not exist, however, for static elements such as textflows.

A terminal element can be a static element 843, a hyperlink element 844, a field element 845 or a page server command element 846, as shown in Figure 27. A static element 843 can be a style element 847 with an associated style object 854, a textflow element 848 with an associated styled text object 855, an image element 849 with an associated image element 856, a graphic element 850 with an associated graphic object 857, a video clip element 851 with an associated video clip object 858, an audio clip element 852 with an associated audio clip object 859, or a script element 853 with an associated script object 860, as shown in Figure 28.

A page instance has a background field 833 which is used to record any digital ink captured on the page which does not apply to a specific input element.

In the preferred form of the invention, a tag map 811 is associated with each page instance to allow tags on the page to be translated into locations on the page.

#### 1.4 THE NETPAGE NETWORK

In a preferred embodiment, a netpage network consists of a distributed set of netpage page servers 10, netpage registration servers 11, netpage ID servers 12, netpage application servers 13, netpage publication servers 14, and netpage printers 601 connected via a network 19 such as the Internet, as shown in Figure 3.

The netpage registration server 11 is a server which records relationships between users, pens, printers, applications and publications, and thereby authorizes various network activities. It authenticates users and acts as a signing proxy on behalf of authenticated users in application transactions. It also provides handwriting recognition services. As described above, a netpage page server 10 maintains persistent information about page descriptions and page instances. The netpage network includes any number of page servers, each handling a subset of page instances. Since a page server also maintains user input values for each page instance, clients such as netpage printers send netpage input directly to the appropriate page server. The page server interprets any such input relative to the description of the corresponding page.

A netpage ID server 12 allocates document IDs 51 on demand, and provides load-balancing of page servers via its ID allocation scheme.

A netpage printer uses the Internet Distributed Name System (DNS), or similar, to resolve a netpage page ID 50 into the network address of the netpage page server handling the corresponding page instance.

A netpage application server 13 is a server which hosts interactive netpage applications. A netpage publication server 14 is an application server which publishes netpage documents to netpage printers. They are described in detail in Section 2.

Netpage servers can be hosted on a variety of network server platforms from manufacturers such as IBM, Hewlett-Packard, and Sun. Multiple netpage servers can run concurrently on a single host, and a single server can be distributed over a number of hosts. Some or all of the functionality provided by netpage servers, and in particular the functionality provided by the ID server and the page server, can also be provided directly

## 1.5 THE NETPAGE PRINTER

Apart from identity and security settings in non-volatile memory, the netpage printer contains no persistent storage. As far as a user is concerned, “the network is the computer”. Netpages function interactively across space and time with the help of the distributed netpage page servers 10, independently of particular netpage printers.

The netpage printer receives subscribed netpage documents from netpage publication servers 14. Each document is distributed in two parts: the page layouts, and the actual text and image objects which populate the pages. Because of personalization, page layouts are typically specific to a particular subscriber and so are pointcast to the subscriber's printer via the appropriate page server. Text and image objects, on the other hand, are typically shared with other subscribers, and so are multicast to all subscribers' printers and the appropriate page servers.

The netpage publication server optimizes the segmentation of document  
20 content into pointcasts and multicasts. After receiving the pointcast of a document's page  
layouts, the printer knows which multicasts, if any, to listen to.

Once the printer has received the complete page layouts and objects that define the document to be printed, it can print the document.

The printer rasterizes and prints odd and even pages simultaneously on both  
25 sides of the sheet. It contains duplexed print engine controllers 760 and print engines  
utilizing Memjet™ printheads 350 for this purpose.

The printing process consists of two decoupled stages: rasterization of page descriptions, and expansion and printing of page images. The raster image processor (RIP) consists of one or more standard DSPs 757 running in parallel. The duplexed print engine controllers consist of custom processors which expand, dither and print page

Printers not enabled for IR printing have the option to print tags using IR-absorptive black ink, although this restricts tags to otherwise empty areas of the page.

A normal netpage printer prints netpages on sheets of paper. More specialised netpage printers may print onto more specialised surfaces, such as globes. Each printer supports at least one surface type, and supports at least one tag tiling scheme, and hence tag map, for each surface type. The tag map 811 which describes the tag tiling scheme actually used to print a document becomes associated with that document so that the document's tags can be correctly interpreted.

15           A preferred embodiment of the netpage printer is described in greater detail in  
Section 6 below, with reference to Figures 11 to 16.

The netpage system can operate using printers made with a wide range of digital printing technologies, including thermal inkjet, piezoelectric inkjet, laser electrophotographic, and others. However, for wide consumer acceptance, it is desirable that a netpage printer have the following characteristics:

- photographic quality color printing
- high quality text printing
- high reliability
- low printer cost
- low ink cost
- low paper cost
- simple operation

- nearly silent printing
  - high printing speed
  - simultaneous double sided printing
  - compact form factor
- 5 • low power consumption

No commercially available printing technology has all of these characteristics.

To enable to production of printers with these characteristics, the present applicant has invented a new print technology, referred to as Memjet™ technology. Memjet™ is a drop-on-demand inkjet technology that incorporates pagewidth printheads  
10 fabricated using microelectromechanical systems (MEMS) technology. Figure 17 shows a single printing element 300 of a Memjet™ printhead. The netpage wallprinter incorporates 168960 printing elements 300 to form a 1600 dpi pagewidth duplex printer. This printer simultaneously prints cyan, magenta, yellow, black, and infrared inks as well as paper conditioner and ink fixative.

15 The printing element 300 is approximately 110 microns long by 32 microns wide. Arrays of these printing elements are formed on a silicon substrate 301 that incorporates CMOS logic, data transfer, timing, and drive circuits (not shown).

Major elements of the printing element 300 are the nozzle 302, the nozzle rim 303, the nozzle chamber 304, the fluidic seal 305, the ink channel rim 306, the lever arm  
20 307, the active actuator beam pair 308, the passive actuator beam pair 309, the active actuator anchor 310, the passive actuator anchor 311, and the ink inlet 312.

The active actuator beam pair 308 is mechanically joined to the passive actuator beam pair 309 at the join 319. Both beams pairs are anchored at their respective anchor points 310 and 311. The combination of elements 308, 309, 310, 311, and 319  
25 form a cantilevered electrothermal bend actuator 320.

Figure 18 shows a small part of an array of printing elements 300, including a cross section 315 of a printing element 300. The cross section 315 is shown without ink, to clearly show the ink inlet 312 that passes through the silicon wafer 301.

Figures 19(a), 19(b) and 19(c) show the operating cycle of a Memjet™ printing



element 300.

Figure 19(a) shows the quiescent position of the ink meniscus 316 prior to printing an ink droplet. Ink is retained in the nozzle chamber by surface tension at the ink meniscus 316 and at the fluidic seal 305 formed between the nozzle chamber 304 and the ink channel rim 306.

While printing, the printhead CMOS circuitry distributes data from the print engine controller to the correct printing element, latches the data, and buffers the data to drive the electrodes 318 of the active actuator beam pair 308. This causes an electrical current to pass through the beam pair 308 for about one microsecond, resulting in Joule heating. The temperature increase resulting from Joule heating causes the beam pair 308 to expand. As the passive actuator beam pair 309 is not heated, it does not expand, resulting in a stress difference between the two beam pairs. This stress difference is partially resolved by the cantilevered end of the electrothermal bend actuator 320 bending towards the substrate 301. The lever arm 307 transmits this movement to the nozzle chamber 304. The nozzle chamber 304 moves about two microns to the position shown in Figure 19(b). This increases the ink pressure, forcing ink 321 out of the nozzle 302, and causing the ink meniscus 316 to bulge. The nozzle rim 303 prevents the ink meniscus 316 from spreading across the surface of the nozzle chamber 304.

As the temperature of the beam pairs 308 and 309 equalizes, the actuator 320 returns to its original position. This aids in the break-off of the ink droplet 317 from the ink 321 in the nozzle chamber, as shown in Figure 19(c). The nozzle chamber is refilled by the action of the surface tension at the meniscus 316.

Figure 20 shows a segment of a printhead 350. In a netpage printer, the length of the printhead is the full width of the paper (typically 210 mm) in the direction 351. The segment shown is 0.4 mm long (about 0.2% of a complete printhead). When printing, the paper is moved past the fixed printhead in the direction 352. The printhead has 6 rows of interdigitated printing elements 300, printing the six colors or types of ink supplied by the ink inlets 312.

To protect the fragile surface of the printhead during operation, a nozzle guard wafer 330 is attached to the printhead substrate 301. For each nozzle 302 there is a corresponding nozzle guard hole 331 through which the ink droplets are fired. To prevent

the nozzle guard holes 331 from becoming blocked by paper fibers or other debris, filtered air is pumped through the air inlets 332 and out of the nozzle guard holes during printing. To prevent ink 321 from drying, the nozzle guard is sealed while the printer is idle.

## 5 1.6 The Netpage Pen

The active sensing device of the netpage system is typically a pen 101, which, using its embedded controller 134, is able to capture and decode IR position tags from a page via an image sensor. The image sensor is a solid-state device provided with an appropriate filter to permit sensing at only near-infrared wavelengths. As described in  
10 more detail below, the system is able to sense when the nib is in contact with the surface, and the pen is able to sense tags at a sufficient rate to capture human handwriting (i.e. at 200 dpi or greater and 100 Hz or faster). Information captured by the pen is encrypted and wirelessly transmitted to the printer (or base station), the printer or base station interpreting the data with respect to the (known) page structure.

15 The preferred embodiment of the netpage pen operates both as a normal marking ink pen and as a non-marking stylus. The marking aspect, however, is not necessary for using the netpage system as a browsing system, such as when it is used as an Internet interface. Each netpage pen is registered with the netpage system and has a unique pen ID 61. Figure 23 shows the netpage pen class diagram, reflecting pen-related  
20 information maintained by a registration server 11 on the netpage network.

When either nib is in contact with a netpage, the pen determines its position and orientation relative to the page. The nib is attached to a force sensor, and the force on the nib is interpreted relative to a threshold to indicate whether the pen is “up” or “down”. This allows a interactive element on the page to be ‘clicked’ by pressing with  
25 the pen nib, in order to request, say, information from a network. Furthermore, the force is captured as a continuous value to allow, say, the full dynamics of a signature to be verified.

The pen determines the position and orientation of its nib on the netpage by imaging, in the infrared spectrum, an area 193 of the page in the vicinity of the nib. It  
30 decodes the nearest tag and computes the position of the nib relative to the tag from the observed perspective distortion on the imaged tag and the known geometry of the pen

5 Pen actions relative to a netpage are captured as a series of strokes. A stroke consists of a sequence of time-stamped pen positions on the page, initiated by a pen-down event and completed by the subsequent pen-up event. A stroke is also tagged with the page ID 50 of the netpage whenever the page ID changes, which, under normal circumstances, is at the commencement of the stroke.

Each pen has a current nib 824. This is the nib last notified by the pen to the system. In the case of the default netpage pen described above, either the marking black ink nib or the non-marking stylus nib is current. Each pen also has a current nib style 825. This is the nib style last associated with the pen by an application, e.g. in response to the user selecting a color from a palette. The default nib style is the nib style associated with the current nib. Strokes captured through a pen are tagged with the current nib style. When the strokes are subsequently reproduced, they are reproduced in the nib style with which they are tagged.

A sequence of captured strokes is referred to as digital ink. Digital ink forms the basis for the digital exchange of drawings and handwriting, for online recognition of handwriting, and for online verification of signatures.

The pen is wireless and transmits digital ink to the netpage printer via a short-range radio link. The transmitted digital ink is encrypted for privacy and security and packetized for efficient transmission, but is always flushed on a pen-up event to ensure timely handling in the printer.

- 5           When the pen is out-of-range of a printer it buffers digital ink in internal memory, which has a capacity of over ten minutes of continuous handwriting. When the pen is once again within range of a printer, it transfers any buffered digital ink.

- A pen can be registered with any number of printers, but because all state data resides in netpages both on paper and on the network, it is largely immaterial which  
10 printer a pen is communicating with at any particular time.

A preferred embodiment of the pen is described in greater detail in Section 6 below, with reference to Figures 8 to 10.

## 1.7       NETPAGE INTERACTION

- The netpage printer 601 receives data relating to a stroke from the pen 101  
15 when the pen is used to interact with a netpage 1. The coded data 3 of the tags 4 is read by the pen when it is used to execute a movement, such as a stroke. The data allows the identity of the particular page and associated interactive element to be determined and an indication of the relative positioning of the pen relative to the page to be obtained. The indicating data is transmitted to the printer, where it resolves, via the DNS, the page ID  
20 50 of the stroke into the network address of the netpage page server 10 which maintains the corresponding page instance 830. It then transmits the stroke to the page server. If the page was recently identified in an earlier stroke, then the printer may already have the address of the relevant page server in its cache. Each netpage consists of a compact page layout maintained persistently by a netpage page server (see below). The page layout  
25 refers to objects such as images, fonts and pieces of text, typically stored elsewhere on the netpage network.

- When the page server receives the stroke from the pen, it retrieves the page description to which the stroke applies, and determines which element of the page description the stroke intersects. It is then able to interpret the stroke in the context of the  
30 type of the relevant element.

5

hyperlink.

### 1.7.1 Hyperlinks

10

15

20

30

The system includes the pen's current selection 826 in a selection hyperlink activation. The system includes the content of the associated form instance 868 in a form hyperlink activation, although if the hyperlink has its "submit delta" attribute set, only input since the last form submission is included. The system includes an effective return path in all hyperlink activations.

A hyperlinked group 866 is a group element 838 which has an associated hyperlink, as shown in Figure 31. When input occurs through any field element in the group, the hyperlink 844 associated with the group is activated. A hyperlinked group can be used to associate hyperlink behavior with a field such as a checkbox. It can also be used, in conjunction with the “submit delta” attribute of a form hyperlink, to provide continuous input to an application. It can therefore be used to support a “blackboard” interaction model, i.e. where input is captured and therefore shared as soon as it occurs.

A form defines a collection of related input fields used to capture a related set of inputs through a printed netpage. A form allows a user to submit one or more parameters to an application software program running on a server.

A form 867 is a group element 838 in the document hierarchy. It ultimately contains a set of terminal field elements 839. A form instance 868 represents a printed instance of a form. It consists of a set of field instances 870 which correspond to the field elements 845 of the form. Each field instance has an associated value 871, whose type depends on the type of the corresponding field element. Each field value records input through a particular printed form instance, i.e. through one or more printed netpages. The form class diagram is shown in Figure 32.

NPX011US

5

particular time period to be excluded from a search.

10

15

20

25

30





|            | Selection | Click           | Submit selection to application   |
|------------|-----------|-----------------|---|
| Form field | Checkbox  | Any mark        | Assign true to field  |
|            | Text      | Handwriting     | Convert digital ink to text; assign text to field   |
|            | Drawing   | Digital ink     | Assign digital ink to field   |
|            | Signature | Signature       | Verify digital ink signature; generate digital signature of form; assign digital signature to field |
| None       | -         | Circumscription | Assign digital ink to current selection   |

The system maintains a current selection for each pen. The selection consists simply of the most recent stroke captured in the background field. The selection is cleared after an inactivity timeout to ensure predictable behavior.

- 5           The raw digital ink captured in every field is retained on the netpage page server and is optionally transmitted with the form data when the form is submitted to the application. This allows the application to interrogate the raw digital ink should it suspect the original conversion, such as the conversion of handwritten text. This can, for example, involve human intervention at the application level for forms which fail certain
- 10 application-specific consistency checks. As an extension to this, the entire background area of a form can be designated as a drawing field. The application can then decide, on the basis of the presence of digital ink outside the explicit fields of the form, to route the form to a human operator, on the assumption that the user may have indicated amendments to the filled-in fields outside of those fields.
- 15           Figure 38 shows a flowchart of the process of handling pen input relative to a netpage. The process consists of receiving (at 884) a stroke from the pen; identifying (at 885) the page instance 830 to which the page ID 50 in the stroke refers; retrieving (at 886) the page description 5; identifying (at 887) a formatted element 839 whose zone 58 the stroke intersects; determining (at 888) whether the formatted element corresponds to
- 20 a field element, and if so appending (at 892) the received stroke to the digital ink of the field value 871, interpreting (at 893) the accumulated digital ink of the field, and determining (at 894) whether the field is part of a hyperlinked group 866 and if so activating (at 895) the associated hyperlink; alternatively determining (at 889) whether the formatted element corresponds to a hyperlink element and if so activating (at 895)
- 25 the corresponding hyperlink; alternatively, in the absence of an input field or hyperlink,

appending (at 890) the received stroke to the digital ink of the background field 833; and copying (at 891) the received stroke to the current selection 826 of the current pen, as maintained by the registration server.

- Figure 38a shows a detailed flowchart of step 893 in the process shown in Figure 38, where the accumulated digital ink of a field is interpreted according to the type of the field. The process consists of determining (at 896) whether the field is a checkbox and (at 897) whether the digital ink represents a checkmark, and if so assigning (at 898) a true value to the field value; alternatively determining (at 899) whether the field is a text field and if so converting (at 900) the digital ink to computer text, with the help of the appropriate registration server, and assigning (at 901) the converted computer text to the field value; alternatively determining (at 902) whether the field is a signature field and if so verifying (at 903) the digital ink as the signature of the pen's owner, with the help of the appropriate registration server, creating (at 904) a digital signature of the contents of the corresponding form, also with the help of the registration server and using the pen owner's private signature key relating to the corresponding application, and assigning (at 905) the digital signature to the field value.

### 1.7.3 Page Server Commands

A page server command is a command which is handled locally by the page server. It operates directly on form, page and document instances.

- A page server command 907 can be a void form command 908, a duplicate form command 909, a reset form command 910, a get form status command 911, a duplicate page command 912, a reset page command 913, a get page status command 914, a duplicate document command 915, a reset document command 916, or a get document status command 917, as shown in Figure 39.

- A void form command voids the corresponding form instance. A duplicate form command voids the corresponding form instance and then produces an active printed copy of the current form instance with field values preserved. The copy contains the same hyperlink transaction IDs as the original, and so is indistinguishable from the original to an application. A reset form command voids the corresponding form instance and then produces an active printed copy of the form instance with field values discarded. A get form status command produces a printed report on the status of the

corresponding form instance, including who published it, when it was printed, for whom it was printed, and the form status of the form instance.

Since a form hyperlink instance contains a transaction ID, the application has to be involved in producing a new form instance. A button requesting a new form instance  
5 is therefore typically implemented as a hyperlink.

A duplicate page command produces a printed copy of the corresponding page instance with the background field value preserved. If the page contains a form or is part of a form, then the duplicate page command is interpreted as a duplicate form command. A reset page command produces a printed copy of the corresponding page instance with  
10 the background field value discarded. If the page contains a form or is part of a form, then the reset page command is interpreted as a reset form command. A get page status command produces a printed report on the status of the corresponding page instance, including who published it, when it was printed, for whom it was printed, and the status of any forms it contains or is part of.

15 The netpage logo which appears on every netpage is usually associated with a duplicate page element.

When a page instance is duplicated with field values preserved, field values are printed in their native form, i.e. a checkmark appears as a standard checkmark graphic, and text appears as typeset text. Only drawings and signatures appear in their original  
20 form, with a signature accompanied by a standard graphic indicating successful signature verification.

A duplicate document command produces a printed copy of the corresponding document instance with background field values preserved. If the document contains any forms, then the duplicate document command duplicates the forms in the same way a  
25 duplicate form command does. A reset document command produces a printed copy of the corresponding document instance with background field values discarded. If the document contains any forms, then the reset document command resets the forms in the same way a reset form command does. A get document status command produces a printed report on the status of the corresponding document instance, including who  
30 published it, when it was printed, for whom it was printed, and the status of any forms it contains.

If the page server command's "on selected" attribute is set, then the command operates on the page identified by the pen's current selection rather than on the page containing the command. This allows a menu of page server commands to be printed. If the target page doesn't contain a page server command element for the designated page server command, then the command is ignored.

An application can provide application-specific handling by embedding the relevant page server command element in a hyperlinked group. The page server activates the hyperlink associated with the hyperlinked group rather than executing the page server command.

A page server command element is hidden if its "hidden" attribute is set. A hidden command element does not have an input zone on a page and so cannot be activated directly by a user. It can, however, be activated via a page server command embedded in a different page, if that page server command has its "on selected" attribute set.

## **1.8 STANDARD FEATURES OF NETPAGES**

In the preferred form, each netpage is printed with the netpage logo at the bottom to indicate that it is a netpage and therefore has interactive properties. The logo also acts as a copy button. In most cases pressing the logo produces a copy of the page. In the case of a form, the button produces a copy of the entire form. And in the case of a secure document, such as a ticket or coupon, the button elicits an explanatory note or advertising page.

The default single-page copy function is handled directly by the relevant netpage page server. Special copy functions are handled by linking the logo button to an application.

## **1.9 USER HELP SYSTEM**

In a preferred embodiment, the netpage printer has a single button labelled "Help". When pressed it elicits a single page of information, including:

- status of printer connection
- status of printer consumables

- The help menu provides a hierarchical manual on how to use the netpage  
5 system.

10           A document function is initiated by simply pressing the button and then touching any page of the document. The status of a document indicates who published it and when, to whom it was delivered, and to whom and when it was subsequently submitted as a form.

- The netpage network directory allows the user to navigate the hierarchy of publications and services on the network. As an alternative, the user can call the netpage network “900” number “yellow pages” and speak to a human operator. The operator can locate the desired document and route it to the user’s printer. Depending on the document type, the publisher or the user pays the small “yellow pages” service fee.

The help page is obviously unavailable if the printer is unable to print. In this  
20 case the “error” light is lit and the user can request remote diagnosis over the network.

In the following description, news is used as a canonical publication example to illustrate personalization mechanisms in the netpage system. Although news is often used in the limited sense of newspaper and newsmagazine news, the intended scope in the present context is wider.

NPX011US

profile. The advertising content is personalized according to the reader's locality and demographic.

## 2.1 EDITORIAL PERSONALIZATION

A subscriber can draw on two kinds of news sources: those that deliver news publications, and those that deliver news streams. While news publications are aggregated and edited by the publisher, news streams are aggregated either by a news publisher or by a specialized news aggregator. News publications typically correspond to traditional newspapers and newsmagazines, while news streams can be many and varied: a "raw" news feed from a news service, a cartoon strip, a freelance writer's column, a friend's bulletin board, or the reader's own e-mail.

The netpage publication server supports the publication of edited news publications as well as the aggregation of multiple news streams. By handling the aggregation and hence the formatting of news streams selected directly by the reader, the server is able to place advertising on pages over which it otherwise has no editorial control.

The subscriber builds a daily newspaper by selecting one or more contributing news publications, and creating a personalized version of each. The resulting daily editions are printed and bound together into a single newspaper. The various members of a household typically express their different interests and tastes by selecting different daily publications and then customizing them.

For each publication, the reader optionally selects specific sections. Some sections appear daily, while others appear weekly. The daily sections available from The New York Times online, for example, include "Page One Plus", "National", "International", "Opinion", "Business", "Arts/Living", "Technology", and "Sports". The set of available sections is specific to a publication, as is the default subset.

The reader can extend the daily newspaper by creating custom sections, each one drawing on any number of news streams. Custom sections might be created for e-mail and friends' announcements ("Personal"), or for monitoring news feeds for specific topics ("Alerts" or "Clippings").

For each section, the reader optionally specifies its size, either qualitatively

(e.g. short, medium, or long), or numerically (i.e. as a limit on its number of pages), and the desired proportion of advertising, either qualitatively (e.g. high, normal, low, none), or numerically (i.e. as a percentage).

5 The reader also optionally expresses a preference for a large number of shorter articles or a small number of longer articles. Each article is ideally written (or edited) in both short and long forms to support this preference.

10 An article may also be written (or edited) in different versions to match the expected sophistication of the reader, for example to provide children's and adults' versions. The appropriate version is selected according to the reader's age. The reader can specify a "reading age" which takes precedence over their biological age.

The articles which make up each section are selected and prioritized by the editors, and each is assigned a useful lifetime. By default they are delivered to all relevant subscribers, in priority order, subject to space constraints in the subscribers' editions.

15 In sections where it is appropriate, the reader may optionally enable collaborative filtering. This is then applied to articles which have a sufficiently long lifetime. Each article which qualifies for collaborative filtering is printed with rating buttons at the end of the article. The buttons can provide an easy choice (e.g. "liked" and "disliked"), making it more likely that readers will bother to rate the article.

20 Articles with high priorities and short lifetimes are therefore effectively considered essential reading by the editors and are delivered to most relevant subscribers.

25 The reader optionally specifies a serendipity factor, either qualitatively (e.g. do or don't surprise me), or numerically. A high serendipity factor lowers the threshold used for matching during collaborative filtering. A high factor makes it more likely that the corresponding section will be filled to the reader's specified capacity. A different serendipity factor can be specified for different days of the week.

The reader also optionally specifies topics of particular interest within a section, and this modifies the priorities assigned by the editors.

30 The speed of the reader's Internet connection affects the quality at which

At a global level, the reader specifies how quantities, dates, times and monetary  
5 values are localized. This involves specifying whether units are imperial or metric, a  
local timezone and time format, and a local currency, and whether the localization  
consist of *in situ* translation or annotation. These preferences are derived from the  
reader's locality by default.

The language in which a news publication is published, and its corresponding text encoding, is a property of the publication and not a preference expressed by the user. However, the netpage system can be configured to provide automatic translation services in various guises.

The personalization of the editorial content directly affects the advertising content, because advertising is typically placed to exploit the editorial context. Travel ads, for example, are more likely to appear in a travel section than elsewhere. The value of the editorial content to an advertiser (and therefore to the publisher) lies in its ability to attract large numbers of readers with the right demographics.

A news publisher's most profitable product is advertising "space", a multi-dimensional entity determined by the publication's geographic coverage, the size of its readership, its readership demographics, and the page area available for advertising.

NPX011US



per-section basis, taking into account the publication's geographic coverage, the section's readership, the size of each reader's section edition, each reader's advertising proportion, and each reader's demographic.

5 In comparison with other media, the netpage system allows the advertising space to be defined in greater detail, and allows smaller pieces of it to be sold separately. It therefore allows it to be sold at closer to its true value.

10 For example, the same advertising "slot" can be sold in varying proportions to several advertisers, with individual readers' pages randomly receiving the advertisement of one advertiser or another, overall preserving the proportion of space sold to each advertiser.

The netpage system allows advertising to be linked directly to detailed product information and online purchasing. It therefore raises the intrinsic value of the advertising space.

15 Because personalization and localization are handled automatically by netpage publication servers, an advertising aggregator can provide arbitrarily broad coverage of both geography and demographics. The subsequent disaggregation is efficient because it is automatic. This makes it more cost-effective for publishers to deal with advertising aggregators than to directly capture advertising. Even though the advertising aggregator is taking a proportion of advertising revenue, publishers may find the change profit-  
20 neutral because of the greater efficiency of aggregation. The advertising aggregator acts as an intermediary between advertisers and publishers, and may place the same advertisement in multiple publications.

25 It is worth noting that ad placement in a netpage publication can be more complex than ad placement in the publication's traditional counterpart, because the publication's advertising space is more complex. While ignoring the full complexities of negotiations between advertisers, advertising aggregators and publishers, the preferred form of the netpage system provides some automated support for these negotiations, including support for automated auctions of advertising space. Automation is particularly desirable for the placement of advertisements which generate small amounts of income,  
30 such as small or highly localized advertisements.

Once placement has been negotiated, the aggregator captures and edits the advertisement and records it on a netpage ad server. Correspondingly, the publisher records the ad placement on the relevant netpage publication server. When the netpage publication server lays out each user's personalized publication, it picks the relevant advertisements from the netpage ad server.

## 2.3 USER PROFILES

### 2.3.1 Information Filtering

The personalization of news and other publications relies on an assortment of user-specific profile information, including:

- publication customizations
- collaborative filtering vectors
- contact details
- presentation preferences

The customization of a publication is typically publication-specific, and so the customization information is maintained by the relevant netpage publication server.

A collaborative filtering vector consists of the user's ratings of a number of news items. It is used to correlate different users' interests for the purposes of making recommendations. Although there are benefits to maintaining a single collaborative filtering vector independently of any particular publication, there are two reasons why it is more practical to maintain a separate vector for each publication: there is likely to be more overlap between the vectors of subscribers to the same publication than between those of subscribers to different publications; and a publication is likely to want to present its users' collaborative filtering vectors as part of the value of its brand, not to be found elsewhere. Collaborative filtering vectors are therefore also maintained by the relevant netpage publication server.

Contact details, including name, street address, ZIP Code, state, country, telephone numbers, are global by nature, and are maintained by a netpage registration server.

Presentation preferences, including those for quantities, dates and times, are

likewise global and maintained in the same way.

The localization of advertising relies on the locality indicated in the user's contact details, while the targeting of advertising relies on personal information such as date of birth, gender, marital status, income, profession, education, or qualitative  
5 derivatives such as age range and income range.

For those users who choose to reveal personal information for advertising purposes, the information is maintained by the relevant netpage registration server. In the absence of such information, advertising can be targeted on the basis of the demographic associated with the user's ZIP or ZIP+4 Code.

10 Each user, pen, printer, application provider and application is assigned its own unique identifier, and the netpage registration server maintains the relationships between them, as shown in Figures 21, 22, 23 and 24. For registration purposes, a publisher is a special kind of application provider, and a publication is a special kind of application.

Each user 800 may be authorized to use any number of printers 802, and each  
15 printer may allow any number of users to use it. Each user has a single default printer (at 66), to which periodical publications are delivered by default, whilst pages printed on demand are delivered to the printer through which the user is interacting. The server keeps track of which publishers a user has authorized to print to the user's default printer. A publisher does not record the ID of any particular printer, but instead resolves  
20 the ID when it is required.

When a user subscribes 808 to a publication 807, the publisher 806 (i.e. application provider 803) is authorized to print to a specified printer or the user's default printer. This authorization can be revoked at any time by the user. Each user may have several pens 801, but a pen is specific to a single user. If a user is authorized to use a  
25 particular printer, then that printer recognizes any of the user's pens.

The pen ID is used to locate the corresponding user profile maintained by a particular netpage registration server, via the DNS in the usual way.

A Web terminal 809 can be authorized to print on a particular netpage printer, allowing Web pages and netpage documents encountered during Web browsing to be  
30 conveniently printed on the nearest netpage printer.

5

10

### 2.3.2 Favorites List

15

### 2.3.3 History List

20

## 2.4 INTELLIGENT PAGE LAYOUT

25

30

and advertising content, such as placing ads for roofing material specifically within the publication because of a special feature on do-it-yourself roofing repairs.

The editorial content selected for the user, including text and associated images and graphics, is then laid out according to various aesthetic rules.

- 5           The entire process, including the selection of ads and the selection of editorial content, must be iterated once the layout has converged, to attempt to more closely achieve the user's stated section size preference. The section size preference can, however, be matched *on average* over time, allowing significant day-to-day variations.

## 2.5       DOCUMENT FORMAT

- 10           Once the document is laid out, it is encoded for efficient distribution and persistent storage on the netpage network.

- The primary efficiency mechanism is the separation of information specific to a single user's edition and information shared between multiple users' editions. The specific information consists of the page layout. The shared information consists of the  
15       objects to which the page layout refers, including images, graphics, and pieces of text.

- A text object contains fully-formatted text represented in the Extensible Markup Language (XML) using the Extensible Stylesheet Language (XSL). XSL provides precise control over text formatting independently of the region into which the text is being set, which in this case is being provided by the layout. The text object  
20       contains embedded language codes to enable automatic translation, and embedded hyphenation hints to aid with paragraph formatting.

          An image object encodes an image in the JPEG 2000 wavelet-based compressed image format. A graphic object encodes a 2D graphic in Scalable Vector Graphics (SVG) format.

- 25           The layout itself consists of a series of placed image and graphic objects, linked textflow objects through which text objects flow, hyperlinks and input fields as described above, and watermark regions. These layout objects are summarized in Table 3. The layout uses a compact format suitable for efficient distribution and storage.

**Table 3 - netpage layout objects**

| Layout object | Attribute               | Format of linked object |
|---------------|-------------------------|-------------------------|
| Image         | Position                | -                       |
|               | Image object ID         | JPEG 2000               |
| Graphic       | Position                | -                       |
|               | Graphic object ID       | SVG                     |
| Textflow      | Textflow ID             | -                       |
|               | Zone                    | -                       |
|               | Optional text object ID | XML/XSL                 |
| Hyperlink     | Type                    | -                       |
|               | Zone                    | -                       |
|               | Application ID, etc.    | -                       |
| Field         | Type                    | -                       |
|               | Meaning                 | -                       |
|               | Zone                    | -                       |
| Watermark     | Zone                    | -                       |

## 2.6 DOCUMENT DISTRIBUTION

- As described above, for purposes of efficient distribution and persistent storage on the netpage network, a user-specific page layout is separated from the shared objects to which it refers.

When a subscribed publication is ready to be distributed, the netpage publication server allocates, with the help of the netpage ID server 12, a unique ID for each page, page instance, document, and document instance.

- The server computes a set of optimized subsets of the shared content and creates a multicast channel for each subset, and then tags each user-specific layout with the names of the multicast channels which will carry the shared content used by that layout. The server then pointcasts each user's layouts to that user's printer via the appropriate page server, and when the pointcasting is complete, multicasts the shared content on the specified channels. After receiving its pointcast, each page server and printer subscribes to the multicast channels specified in the page layouts. During the multicasts, each page server and printer extracts from the multicast streams those objects referred to by its page layouts. The page servers persistently archive the received page layouts and shared content.

- Once a printer has received all the objects to which its page layouts refer, the

Under normal circumstances, the printer prints pages faster than they can be delivered. Assuming a quarter of each page is covered with images, the average page has a size of less than 400KB. The printer can therefore hold in excess of 100 such pages in its internal 64MB memory, allowing for temporary buffers etc. The printer prints at a rate of one page per second. This is equivalent to 400KB or about 3Mbit of page data per second, which is similar to the highest expected rate of page data delivery over a broadband network.

However, if the printer's internal memory does fill up, then the printer will be unable to make use of a multicast when it first occurs. The netpage publication server therefore allows printers to submit requests for re-multicasts. When a critical number of requests is received or a timeout occurs, the server re-multicasts the corresponding shared objects.

## 2.7 ON-DEMAND DOCUMENTS

When a non-netpage document is requested on demand, it is not personalized, and it is delivered via a designated netpage formatting server which reformats it as a netpage document. A netpage formatting server is a special instance of a netpage publication server. The netpage formatting server has knowledge of various Internet document formats, including Adobe's Portable Document Format (PDF), and Hypertext Markup Language (HTML). In the case of HTML, it can make use of the higher resolution of the printed page to present Web pages in a multi-column format, with a table of contents. It can automatically include all Web pages directly linked to the

requested page. The user can tune this behavior via a preference.

The netpage formatting server makes standard netpage behavior, including interactivity and persistence, available on any Internet document, no matter what its origin and format. It hides knowledge of different document formats from both the netpage printer and the netpage page server, and hides knowledge of the netpage system from Web servers.

### **3 SECURITY**

#### **3.1 CRYPTOGRAPHY**

Cryptography is used to protect sensitive information, both in storage and in transit, and to authenticate parties to a transaction. There are two classes of cryptography in widespread use: secret-key cryptography and public-key cryptography. The netpage network uses both classes of cryptography.

Secret-key cryptography, also referred to as symmetric cryptography, uses the same key to encrypt and decrypt a message. Two parties wishing to exchange messages must first arrange to securely exchange the secret key.

Public-key cryptography, also referred to as asymmetric cryptography, uses two encryption keys. The two keys are mathematically related in such a way that any message encrypted using one key can only be decrypted using the other key. One of these keys is then published, while the other is kept private. The public key is used to encrypt any message intended for the holder of the private key. Once encrypted using the public key, a message can only be decrypted using the private key. Thus two parties can securely exchange messages without first having to exchange a secret key. To ensure that the private key is secure, it is normal for the holder of the private key to generate the key pair.

Public-key cryptography can be used to create a digital signature. The holder of the private key can create a known hash of a message and then encrypt the hash using the private key. Anyone can then verify that the encrypted hash constitutes the “signature” of the holder of the private key with respect to that particular message by decrypting the encrypted hash using the public key and verifying the hash against the message. If the signature is appended to the message, then the recipient of the message can verify both



that the message is genuine and that it has not been altered in transit.

To make public-key cryptography work, there has to be a way to distribute public keys which prevents impersonation. This is normally done using certificates and certificate authorities. A certificate authority is a trusted third party which authenticates the connection between a public key and someone's identity. The certificate authority verifies the person's identity by examining identity documents, and then creates and signs a digital certificate containing the person's identity details and public key. Anyone who trusts the certificate authority can use the public key in the certificate with a high degree of certainty that it is genuine. They just have to verify that the certificate has indeed been signed by the certificate authority, whose public key is well-known.

In most transaction environments, public-key cryptography is only used to create digital signatures and to securely exchange secret session keys. Secret-key cryptography is used for all other purposes.

In the following discussion, when reference is made to the *secure* transmission of information between a netpage printer and a server, what actually happens is that the printer obtains the server's certificate, authenticates it with reference to the certificate authority, uses the public key-exchange key in the certificate to exchange a secret session key with the server, and then uses the secret session key to encrypt the message data. A *session* key, by definition, can have an arbitrarily short lifetime.

### 3.2 NETPAGE PRINTER SECURITY

Each netpage printer is assigned a pair of unique identifiers at time of manufacture which are stored in read-only memory in the printer and in the netpage registration server database. The first ID 62 is public and uniquely identifies the printer on the netpage network. The second ID is secret and is used when the printer is first registered on the network.

When the printer connects to the netpage network for the first time after installation, it creates a signature public/private key pair. It transmits the secret ID and the public key securely to the netpage registration server. The server compares the secret ID against the printer's secret ID recorded in its database, and accepts the registration if the IDs match. It then creates and signs a certificate containing the printer's public ID

and public signature key, and stores the certificate in the registration database.

The netpage registration server acts as a certificate authority for netpage printers, since it has access to secret information allowing it to verify printer identity.

When a user subscribes to a publication, a record is created in the netpage registration server database authorizing the publisher to print the publication to the user's default printer or a specified printer. Every document sent to a printer via a page server is addressed to a particular user and is signed by the publisher using the publisher's private signature key. The page server verifies, via the registration database, that the publisher is authorized to deliver the publication to the specified user. The page server verifies the signature using the publisher's public key, obtained from the publisher's certificate stored in the registration database.

The netpage registration server accepts requests to add printing authorizations to the database, so long as those requests are initiated via a pen registered to the printer.

### 3.3 NETPAGE PEN SECURITY

Each netpage pen is assigned a unique identifier at time of manufacture which is stored in read-only memory in the pen and in the netpage registration server database. The pen ID 61 uniquely identifies the pen on the netpage network.

A netpage pen can "know" a number of netpage printers, and a printer can "know" a number of pens. A pen communicates with a printer via a radio frequency signal whenever it is within range of the printer. Once a pen and printer are registered, they regularly exchange session keys. Whenever the pen transmits digital ink to the printer, the digital ink is always encrypted using the appropriate session key. Digital ink is never transmitted in the clear.

A pen stores a session key for every printer it knows, indexed by printer ID, and a printer stores a session key for every pen it knows, indexed by pen ID. Both have a large but finite storage capacity for session keys, and will forget a session key on a least-recently-used basis if necessary.

When a pen comes within range of a printer, the pen and printer discover whether they know each other. If they don't know each other, then the printer determines whether it is supposed to know the pen. This might be, for example, because the pen

belongs to a user who is registered to use the printer. If the printer is meant to know the pen but doesn't, then it initiates the automatic pen registration procedure. If the printer isn't meant to know the pen, then it agrees with the pen to ignore it until the pen is placed in a charging cup, at which time it initiates the registration procedure.

5           In addition to its public ID, the pen contains a secret key-exchange key. The key-exchange key is also recorded in the netpage registration server database at time of manufacture. During registration, the pen transmits its pen ID to the printer, and the printer transmits the pen ID to the netpage registration server. The server generates a session key for the printer and pen to use, and securely transmits the session key to the  
10 printer. It also transmits a copy of the session key encrypted with the pen's key-exchange key. The printer stores the session key internally, indexed by the pen ID, and transmits the encrypted session key to the pen. The pen stores the session key internally, indexed by the printer ID.

          Although a fake pen can impersonate a pen in the pen registration protocol,  
15 only a real pen can decrypt the session key transmitted by the printer.

          When a previously unregistered pen is first registered, it is of limited use until it is linked to a user. A registered but "un-owned" pen is only allowed to be used to request and fill in netpage user and pen registration forms, to register a new user to which the new pen is automatically linked, or to add a new pen to an existing user.

20           The pen uses secret-key rather than public-key encryption because of hardware performance constraints in the pen.

### **3.4       SECURE DOCUMENTS**

          The netpage system supports the delivery of secure documents such as tickets and coupons. The netpage printer includes a facility to print watermarks, but will only do  
25 so on request from publishers who are suitably authorized. The publisher indicates its authority to print watermarks in its certificate, which the printer is able to authenticate.

          The "watermark" printing process uses an alternative dither matrix in specified "watermark" regions of the page. Back-to-back pages contain mirror-image watermark regions which coincide when printed. The dither matrices used in odd and even pages'  
30 watermark regions are designed to produce an interference effect when the regions are

viewed together, achieved by looking *through* the printed sheet.

The effect is similar to a watermark in that it is not visible when looking at only one side of the page, and is lost when the page is copied by normal means.

Pages of secure documents cannot be copied using the built-in netpage copy  
5 mechanism described in Section 1.9 above. This extends to copying netpages on  
netpage-aware photocopiers.

Secure documents are typically generated as part of e-commerce transactions.  
They can therefore include the user's photograph which was captured when the user  
registered biometric information with the netpage registration server, as described in  
10 Section 2.

When presented with a secure netpage document, the recipient can verify its  
authenticity by requesting its status in the usual way. The unique ID of a secure  
document is only valid for the lifetime of the document, and secure document IDs are  
allocated non-contiguously to prevent their prediction by opportunistic forgers. A secure  
15 document verification pen can be developed with built-in feedback on verification  
failure, to support easy point-of-presentation document verification.

Clearly neither the watermark nor the user's photograph are secure in a  
cryptographic sense. They simply provide a significant obstacle to casual forgery. Online  
document verification, particularly using a verification pen, provides an added level of  
20 security where it is needed, but is still not entirely immune to forgeries.

### **3.5 NON-REPUDIATION**

In the netpage system, forms submitted by users are delivered reliably to forms  
handlers and are persistently archived on netpage page servers. It is therefore impossible  
for recipients to repudiate delivery.

25 E-commerce payments made through the system, as described in Section 4, are  
also impossible for the payee to repudiate.

## **4 ELECTRONIC COMMERCE MODEL**

### **4.1 SECURE ELECTRONIC TRANSACTION (SET)**

The netpage system uses the Secure Electronic Transaction (SET) system as

In SET, cardholders and merchants register with a certificate authority and are issued with certificates containing their public signature keys. The certificate authority verifies a cardholder's registration details with the card issuer as appropriate, and verifies a merchant's registration details with the acquirer as appropriate. Cardholders and merchants store their respective private signature keys securely on their computers. During the payment process, these certificates are used to mutually authenticate a merchant and cardholder, and to authenticate them both to the payment gateway.

## 15 4.2 SET PAYMENTS

The netpage system uses biometrics to authenticate the user and authorize SET payments. Because the system is pen-based, the biometric used is the user's on-line signature, consisting of time-varying pen position and pressure. A fingerprint biometric can also be used by designing a fingerprint sensor into the pen, although at a higher cost. The type of biometric used only affects the capture of the biometric, not the authorization aspects of the system.

NPX011US

information is also stored, giving the netpage registration server enough information to act as the user's proxy in any SET payment transaction.

When the user eventually supplies the biometric to complete a payment, for example by signing a netpage order form, the printer securely transmits the order information, the pen ID and the biometric data to the netpage registration server. The server verifies the biometric with respect to the user identified by the pen ID, and from then on acts as the user's proxy in completing the SET payment transaction.

#### 4.3 MICRO-PAYMENTS

The netpage system includes a mechanism for micro-payments, to allow the user to be conveniently charged for printing low-cost documents on demand and for copying copyright documents, and possibly also to allow the user to be reimbursed for expenses incurred in printing advertising material. The latter depends on the level of subsidy already provided to the user.

When the user registers for e-commerce, a network account is established which aggregates micro-payments. The user receives a statement on a regular basis, and can settle any outstanding debit balance using the standard payment mechanism.

The network account can be extended to aggregate subscription fees for periodicals, which would also otherwise be presented to the user in the form of individual statements.

#### 4.4 TRANSACTIONS

When a user requests a netpage in a particular application context, the application is able to embed a user-specific transaction ID in the page. Subsequent input through the page is tagged with the transaction ID, and the application is thereby able to establish an appropriate context for the user's input.

When input occurs through a page which is not user-specific, however, the application must use the user's unique identity to establish a context. A typical example involves adding items from a pre-printed catalog page to the user's virtual "shopping cart". To protect the user's privacy, however, the unique user ID known to the netpage system is not divulged to applications. This is to prevent different application providers from easily correlating independently accumulated behavioral data.

5

**user.**

10

15

debit the provider for click-through fees etc.

20

## 4.5

Figure 40.

25

30

automatically transfers copyright fees from users to rights holders when users print copyright content.

## **5 COMMUNICATIONS PROTOCOLS**

A communications protocol defines an ordered exchange of messages between entities. In the netpage system, entities such as pens, printers and servers utilise a set of defined protocols to cooperatively handle user interaction with the netpage system.

Each protocol is illustrated by way of a sequence diagram in which the horizontal dimension is used to represent message flow and the vertical dimension is used to represent time. Each entity is represented by a rectangle containing the name of the entity and a vertical column representing the lifeline of the entity. During the time an entity exists, the lifeline is shown as a dashed line. During the time an entity is active, the lifeline is shown as a double line. Because the protocols considered here do not create or destroy entities, lifelines are generally cut short as soon as an entity ceases to participate in a protocol.

### **5.1 SUBSCRIPTION DELIVERY PROTOCOL**

A preferred embodiment of a subscription delivery protocol is shown in Figure 43.

A large number of users may subscribe to a periodical publication. Each user's edition may be laid out differently, but many users' editions will share common content such as text objects and image objects. The subscription delivery protocol therefore delivers document structures to individual printers via pointcast, but delivers shared content objects via multicast.

The application (i.e. publisher) first obtains a document ID 51 for each document from an ID server 12. It then sends each document structure, including its document ID and page descriptions, to the page server 10 responsible for the document's newly allocated ID. It includes its own application ID 64, the subscriber's alias ID 65, and the relevant set of multicast channel names. It signs the message using its private signature key.

The page server uses the application ID and alias ID to obtain from the registration server the corresponding user ID 60, the user's selected printer ID 62 (which



may be explicitly selected for the application, or may be the user's default printer), and the application's certificate.

The application's certificate allows the page server to verify the message signature. The page server's request to the registration server fails if the application ID  
5 and alias ID don't together identify a subscription 808.

The page server then allocates document and page instance IDs and forwards the page descriptions, including page IDs 50, to the printer. It includes the relevant set of multicast channel names for the printer to listen to.

It then returns the newly allocated page IDs to the application for future  
10 reference.

Once the application has distributed all of the document structures to the subscribers' selected printers via the relevant page servers, it multicasts the various subsets of the shared objects on the previously selected multicast channels. Both page servers and printers monitor the appropriate multicast channels and receive their required  
15 content objects. They are then able to populate the previously pointcast document structures. This allows the page servers to add complete documents to their databases, and it allows the printers to print the documents.

## **5.2 HYPERLINK ACTIVATION PROTOCOL**

A preferred embodiment of a hyperlink activation protocol is shown in Figure  
20 45.

When a user clicks on a netpage with a netpage pen, the pen communicates the click to the nearest netpage printer 601. The click identifies the page and a location on the page. The printer already knows the ID 61 of the pen from the pen connection protocol.

The printer determines, via the DNS, the network address of the page server  
25 10a handling the particular page ID 50. The address may already be in its cache if the user has recently interacted with the same page. The printer then forwards the pen ID, its own printer ID 62, the page ID and click location to the page server.

The page server loads the page description 5 identified by the page ID and  
30 determines which input element's zone 58, if any, the click lies in. Assuming the

relevant input element is a hyperlink element 844, the page server then obtains the associated application ID 64 and link ID 54, and determines, via the DNS, the network address of the application server hosting the application 71.

5 The page server uses the pen ID 61 to obtain the corresponding user ID 60 from the registration server 11, and then allocates a globally unique hyperlink request ID 52 and builds a hyperlink request 934. The hyperlink request class diagram is shown in Figure 44. The hyperlink request records the IDs of the requesting user and printer, and identifies the clicked hyperlink instance 862. The page server then sends its own server ID 53, the hyperlink request ID, and the link ID to the application.

10 The application produces a response document according to application-specific logic, and obtains a document ID 51 from an ID server 12. It then sends the document to the page server 10b responsible for the document's newly allocated ID, together with the requesting page server's ID and the hyperlink request ID.

15 The second page server sends the hyperlink request ID and application ID to the first page server to obtain the corresponding user ID and printer ID 62. The first page server rejects the request if the hyperlink request has expired or is for a different application.

20 The second page server allocates document instance and page IDs 50, returns the newly allocated page IDs to the application, adds the complete document to its own database, and finally sends the page descriptions to the requesting printer.

The hyperlink instance may include a meaningful transaction ID 55, in which case the first page server includes the transaction ID in the message sent to the application. This allows the application to establish a transaction-specific context for the hyperlink activation.

25 If the hyperlink requires a user alias, i.e. its "alias required" attribute is set, then the first page server sends both the pen ID 61 and the hyperlink's application ID 64 to the registration server 11 to obtain not just the user ID corresponding to the pen ID but also the alias ID 65 corresponding to the application ID and the user ID. It includes the alias ID in the message sent to the application, allowing the application to establish a  
30 user-specific context for the hyperlink activation.

### 5.3 HANDWRITING RECOGNITION PROTOCOL

When a user draws a stroke on a netpage with a netpage pen, the pen communicates the stroke to the nearest netpage printer. The stroke identifies the page and a path on the page.

- 5           The printer forwards the pen ID 61, its own printer ID 62, the page ID 50 and stroke path to the page server 10 in the usual way.

          The page server loads the page description 5 identified by the page ID and determines which input element's zone 58, if any, the stroke intersects. Assuming the relevant input element is a text field 878, the page server appends the stroke to the text  
10   field's digital ink.

          After a period of inactivity in the zone of the text field, the page server sends the pen ID and the pending strokes to the registration server 11 for interpretation. The registration server identifies the user corresponding to the pen, and uses the user's accumulated handwriting model 822 to interpret the strokes as handwritten text. Once it  
15   has converted the strokes to text, the registration server returns the text to the requesting page server. The page server appends the text to the text value of the text field.

### 5.4 SIGNATURE VERIFICATION PROTOCOL

Assuming the input element whose zone the stroke intersects is a signature field 880, the page server 10 appends the stroke to the signature field's digital ink.

- 20           After a period of inactivity in the zone of the signature field, the page server sends the pen ID 61 and the pending strokes to the registration server 11 for verification. It also sends the application ID 64 associated with the form of which the signature field is part, as well as the form ID 56 and the current data content of the form. The registration server identifies the user corresponding to the pen, and uses the user's  
25   dynamic signature biometric 818 to verify the strokes as the user's signature. Once it has verified the signature, the registration server uses the application ID 64 and user ID 60 to identify the user's application-specific private signature key. It then uses the key to generate a digital signature of the form data, and returns the digital signature to the requesting page server. The page server assigns the digital signature to the signature field  
30   and sets the associated form's status to frozen.

## 5.5 FORM SUBMISSION PROTOCOL

5 Form submission occurs via a form hyperlink activation. It thus follows the protocol defined in Section 5.2, with some form-specific additions.

## 5.6 COMMISSION PAYMENT PROTOCOL

15 In an e-commerce environment, fees and commissions may be payable from an application provider to a publisher on click-throughs, transactions and sales. Commissions on fees and commissions on commissions may also be payable from the publisher to the provider of the printer.

The target application receives the hyperlink request ID from the page server 10 when the hyperlink is first activated, as described in Section 5.2. When the target application needs to credit the source application provider, it sends the application provider credit to the original page server together with the hyperlink request ID. The page server uses the hyperlink request ID to identify the source application, and sends the credit on to the relevant registration server 11 together with the source application ID 64, its own server ID 53, and the hyperlink request ID. The registration server credits the corresponding application provider's account 827. It also notifies the application

provider.

If the application provider needs to credit the printer provider, it sends the printer provider credit to the original page server together with the hyperlink request ID. The page server uses the hyperlink request ID to identify the printer, and sends the credit  
5 on to the relevant registration server together with the printer ID. The registration server credits the corresponding printer provider account 814.

The source application provider is optionally notified of the identity of the target application provider, and the printer provider of the identity of the source application provider.

## 10 6. NETPAGE PEN DESCRIPTION

### 6.1 PEN MECHANICS

Referring to Figures 8 and 9, the pen, generally designated by reference numeral 101, includes a housing 102 in the form of a plastics moulding having walls 103 defining an interior space 104 for mounting the pen components. The pen top 105 is in  
15 operation rotatably mounted at one end 106 of the housing 102. A semi-transparent cover 107 is secured to the opposite end 108 of the housing 102. The cover 107 is also of moulded plastics, and is formed from semi-transparent material in order to enable the user to view the status of the LED mounted within the housing 102. The cover 107 includes a main part 109 which substantially surrounds the end 108 of the housing 102  
20 and a projecting portion 110 which projects back from the main part 109 and fits within a corresponding slot 111 formed in the walls 103 of the housing 102. A radio antenna 112 is mounted behind the projecting portion 110, within the housing 102. Screw threads 113 surrounding an aperture 113A on the cover 107 are arranged to receive a metal end piece 114, including corresponding screw threads 115. The metal end piece 114 is  
25 removable to enable ink cartridge replacement.

Also mounted within the cover 107 is a tri-color status LED 116 on a flex PCB 117. The antenna 112 is also mounted on the flex PCB 117. The status LED 116 is mounted at the top of the pen 101 for good all-around visibility.

The pen can operate both as a normal marking ink pen and as a non-marking  
30 stylus. An ink pen cartridge 118 with nib 119 and a stylus 120 with stylus nib 121 are

mounted side by side within the housing 102. Either the ink cartridge nib 119 or the stylus nib 121 can be brought forward through open end 122 of the metal end piece 114, by rotation of the pen top 105. Respective slider blocks 123 and 124 are mounted to the ink cartridge 118 and stylus 120, respectively. A rotatable cam barrel 125 is secured to the pen top 105 in operation and arranged to rotate therewith. The cam barrel 125 includes a cam 126 in the form of a slot within the walls 181 of the cam barrel. Cam followers 127 and 128 projecting from slider blocks 123 and 124 fit within the cam slot 126. On rotation of the cam barrel 125, the slider blocks 123 or 124 move relative to each other to project either the pen nib 119 or stylus nib 121 out through the hole 122 in the metal end piece 114. The pen 101 has three states of operation. By turning the top 105 through 90° steps, the three states are:

- Stylus 120 nib 121 out;
- Ink cartridge 118 nib 119 out; and
- Neither ink cartridge 118 nib 119 out nor stylus 120 nib 121 out.

A second flex PCB 129, is mounted on an electronics chassis 130 which sits within the housing 102. The second flex PCB 129 mounts an infrared LED 131 for providing infrared radiation for projection onto the surface. An image sensor 132 is provided mounted on the second flex PCB 129 for receiving reflected radiation from the surface. The second flex PCB 129 also mounts a radio frequency chip 133, which includes an RF transmitter and RF receiver, and a controller chip 134 for controlling operation of the pen 101. An optics block 135 (formed from moulded clear plastics) sits within the cover 107 and projects an infrared beam onto the surface and receives images onto the image sensor 132. Power supply wires 136 connect the components on the second flex PCB 129 to battery contacts 137 which are mounted within the cam barrel 125. A terminal 138 connects to the battery contacts 137 and the cam barrel 125. A three volt rechargeable battery 139 sits within the cam barrel 125 in contact with the battery contacts. An induction charging coil 140 is mounted about the second flex PCB 129 to enable recharging of the battery 139 via induction. The second flex PCB 129 also mounts an infrared LED 143 and infrared photodiode 144 for detecting displacement in the cam barrel 125 when either the stylus 120 or the ink cartridge 118 is used for writing, in order to enable a determination of the force being applied to the surface by the pen nib

119 or stylus nib 121. The IR photodiode 144 detects light from the IR LED 143 via reflectors (not shown) mounted on the slider blocks 123 and 124.

Rubber grip pads 141 and 142 are provided towards the end 108 of the housing 102 to assist gripping the pen 101, and top 105 also includes a clip 142 for clipping the pen 101 to a pocket.

## 6.2 PEN CONTROLLER

The pen 101 is arranged to determine the position of its nib (stylus nib 121 or ink cartridge nib 119) by imaging, in the infrared spectrum, an area of the surface in the vicinity of the nib. It records the location data from the nearest location tag, and is arranged to calculate the distance of the nib 121 or 119 from the location tag utilising optics 135 and controller chip 134. The controller chip 134 calculates the orientation of the pen and the nib-to-tag distance from the perspective distortion observed on the imaged tag.

Utilising the RF chip 133 and antenna 112 the pen 101 can transmit the digital ink data (which is encrypted for security and packaged for efficient transmission) to the computing system.

When the pen is in range of a receiver, the digital ink data is transmitted as it is formed. When the pen 101 moves out of range, digital ink data is buffered within the pen 101 (the pen 101 circuitry includes a buffer arranged to store digital ink data for approximately 12 minutes of the pen motion on the surface) and can be transmitted later.

The controller chip 134 is mounted on the second flex PCB 129 in the pen 101. Figure 10 is a block diagram illustrating in more detail the architecture of the controller chip 134. Figure 10 also shows representations of the RF chip 133, the image sensor 132, the tri-color status LED 116, the IR illumination LED 131, the IR force sensor LED 143, and the force sensor photodiode 144.

The pen controller chip 134 includes a controlling processor 145. Bus 146 enables the exchange of data between components of the controller chip 134. Flash memory 147 and a 512 KB DRAM 148 are also included. An analog-to-digital converter 149 is arranged to convert the analog signal from the force sensor photodiode 144 to a digital signal.

An image sensor interface 152 interfaces with the image sensor 132. A transceiver controller 153 and base band circuit 154 are also included to interface with the RF chip 133 which includes an RF circuit 155 and RF resonators and inductors 156 connected to the antenna 112.

5           The controlling processor 145 captures and decodes location data from tags from the surface via the image sensor 132, monitors the force sensor photodiode 144, controls the LEDs 116, 131 and 143, and handles short-range radio communication via the radio transceiver 153. It is a medium-performance (~40MHz) general-purpose RISC processor.

10           The processor 145, digital transceiver components (transceiver controller 153 and baseband circuit 154), image sensor interface 152, flash memory 147 and 512KB DRAM 148 are integrated in a single controller ASIC. Analog RF components (RF circuit 155 and RF resonators and inductors 156) are provided in the separate RF chip.

15           The image sensor is a 215x215 pixel CCD (such a sensor is produced by Matsushita Electronic Corporation, and is described in a paper by Itakura, K T Nobusada, N Okusanya, R Nagayoshi, and M Ozaki, "A 1mm 50k-Pixel IT CCD Image Sensor for Miniature Camera System", IEEE Transactions on Electronic Devices, Volt 47, number 1, January 2000, which is incorporated herein by reference) with an IR filter.

20           The controller ASIC 134 enters a quiescent state after a period of inactivity when the pen 101 is not in contact with a surface. It incorporates a dedicated circuit 150 which monitors the force sensor photodiode 144 and wakes up the controller 134 via the power manager 151 on a pen-down event.

25           The radio transceiver communicates in the unlicensed 900MHz band normally used by cordless telephones, or alternatively in the unlicensed 2.4GHz industrial, scientific and medical (ISM) band, and uses frequency hopping and collision detection to provide interference-free communication.

          In an alternative embodiment, the pen incorporates an Infrared Data Association (IrDA) interface for short-range communication with a base station or netpage printer.

30           In a further embodiment, the pen 101 includes a pair of orthogonal





printer via a hose 639. This provides filtered air to the printheads to prevent ingress of micro particles into the Memjet™ printheads 350 which might otherwise clog the printhead nozzles. By incorporating the air filter within the cartridge, the operational life of the filter is effectively linked to the life of the cartridge. The ink cartridge is a fully recyclable product with a capacity for printing and gluing 3000 pages (1500 sheets).

Referring to Figure 12, the motorized media pick-up roller assembly 626 pushes the top sheet directly from the media tray past a paper sensor on the first print engine 602 into the duplexed Memjet™ printhead assembly. The two Memjet™ print engines 602 and 603 are mounted in an opposing in-line sequential configuration along the straight paper path. The paper 604 is drawn into the first print engine 602 by integral, powered pick-up rollers 626. The position and size of the paper 604 is sensed and full bleed printing commences. Fixative is printed simultaneously to aid drying in the shortest possible time.

The paper exits the first Memjet™ print engine 602 through a set of powered exit spike wheels (aligned along the straight paper path), which act against a rubberized roller. These spike wheels contact the 'wet' printed surface and continue to feed the sheet 604 into the second Memjet™ print engine 603.

Referring to Figures 12 and 12a, the paper 604 passes from the duplexed print engines 602 and 603 into the binder assembly 605. The printed page passes between a powered spike wheel axle 670 with a fibrous support roller and another movable axle with spike wheels and a momentary action glue wheel. The movable axle/glue assembly 673 is mounted to a metal support bracket and it is transported forward to interface with the powered axle 670 via gears by action of a camshaft. A separate motor powers this camshaft.

The glue wheel assembly 673 consists of a partially hollow axle 679 with a rotating coupling for the glue supply hose 641 from the ink cartridge 627. This axle 679 connects to a glue wheel, which absorbs adhesive by capillary action through radial holes. A molded housing 682 surrounds the glue wheel, with an opening at the front. Pivoting side moldings and sprung outer doors are attached to the metal bracket and hinge out sideways when the rest of the assembly 673 is thrust forward. This action exposes the glue wheel through the front of the molded housing 682. Tension springs

close the assembly and effectively cap the glue wheel during periods of inactivity.

As the sheet 604 passes into the glue wheel assembly 673, adhesive is applied to one vertical edge on the front side (apart from the first sheet of a document) as it is transported down into the binding assembly 605.

## 5 7.2 PRINTER CONTROLLER ARCHITECTURE

The netpage printer controller consists of a controlling processor 750, a factory-installed or field-installed network interface module 625, a radio transceiver (transceiver controller 753, baseband circuit 754, RF circuit 755, and RF resonators and inductors 756), dual raster image processor (RIP) DSPs 757, duplexed print engine controllers 10 760a and 760b, flash memory 658, and 64MB of DRAM 657, as illustrated in Figure 14.

The controlling processor handles communication with the network 19 and with local wireless netpage pens 101, senses the help button 617, controls the user interface LEDs 613-616, and feeds and synchronizes the RIP DSPs 757 and print engine controllers 760. It consists of a medium-performance general-purpose microprocessor. 15 The controlling processor 750 communicates with the print engine controllers 760 via a high-speed serial bus 659.

The RIP DSPs rasterize and compress page descriptions to the netpage printer's compressed page format. Each print engine controller expands, dithers and prints page images to its associated Memjet™ printhead 350 in real time (i.e. at over 30 pages per 20 minute). The duplexed print engine controllers print both sides of a sheet simultaneously.

The master print engine controller 760a controls the paper transport and monitors ink usage in conjunction with the master QA chip 665 and the ink cartridge QA chip 761.

The printer controller's flash memory 658 holds the software for both the 25 processor 750 and the DSPs 757, as well as configuration data. This is copied to main memory 657 at boot time.

The processor 750, DSPs 757, and digital transceiver components (transceiver controller 753 and baseband circuit 754) are integrated in a single controller ASIC 656. Analog RF components (RF circuit 755 and RF resonators and inductors 756) are 30 provided in a separate RF chip 762. The network interface module 625 is separate, since

netpage printers allow the network connection to be factory-selected or field-selected. Flash memory 658 and the 2×256Mbit (64MB) DRAM 657 is also off-chip. The print engine controllers 760 are provided in separate ASICs.

A variety of network interface modules 625 are provided, each providing a netpage network interface 751 and optionally a local computer or network interface 752. Netpage network Internet interfaces include POTS modems, Hybrid Fiber-Coax (HFC) cable modems, ISDN modems, DSL modems, satellite transceivers, current and next-generation cellular telephone transceivers, and wireless local loop (WLL) transceivers. Local interfaces include IEEE 1284 (parallel port), 10Base-T and 100Base-T Ethernet, USB and USB 2.0, IEEE 1394 (Firewire), and various emerging home networking interfaces. If an Internet connection is available on the local network, then the local network interface can be used as the netpage network interface.

The radio transceiver 753 communicates in the unlicensed 900MHz band normally used by cordless telephones, or alternatively in the unlicensed 2.4GHz industrial, scientific and medical (ISM) band, and uses frequency hopping and collision  
15 detection to provide interference-free communication.

The printer controller optionally incorporates an Infrared Data Association (IrDA) interface for receiving data “squirted” from devices such as netpage cameras. In an alternative embodiment, the printer uses the IrDA interface for short-range communication with suitably configured netpage pens.

### 7.2.1 RASTERIZATION AND PRINTING

Once the main processor 750 has received and verified the document's page layouts and page objects, it runs the appropriate RIP software on the DSPs 757.

The DSPs 757 rasterize each page description and compress the rasterized page  
25 image. The main processor stores each compressed page image in memory. The simplest  
way to load-balance multiple DSPs is to let each DSP rasterize a separate page. The  
DSPs can always be kept busy since an arbitrary number of rasterized pages can, in  
general, be stored in memory. This strategy only leads to potentially poor DSP utilization  
when rasterizing short documents.

30 Watermark regions in the page description are rasterized to a contone-

resolution bi-level bitmap which is losslessly compressed to negligible size and which forms part of the compressed page image.

The infrared (IR) layer of the printed page contains coded netpage tags at a density of about six per inch. Each tag encodes the page ID, tag ID, and control bits, and  
5 the data content of each tag is generated during rasterization and stored in the compressed page image.

The main processor 750 passes back-to-back page images to the duplexed print engine controllers 760. Each print engine controller 760 stores the compressed page image in its local memory, and starts the page expansion and printing pipeline. Page  
10 expansion and printing is pipelined because it is impractical to store an entire 114MB bi-level CMYK+IR page image in memory.

### 7.2.2 PRINT ENGINE CONTROLLER

The page expansion and printing pipeline of the print engine controller 760 consists of a high speed IEEE 1394 serial interface 659, a standard JPEG decoder 763, a  
15 standard Group 4 Fax decoder 764, a custom halftoner/compositor unit 765, a custom tag encoder 766, a line loader/formatter unit 767, and a custom interface 768 to the Memjet™ printhead 350.

The print engine controller 360 operates in a double buffered manner. While one page is loaded into DRAM 769 via the high speed serial interface 659, the  
20 previously loaded page is read from DRAM 769 and passed through the print engine controller pipeline. Once the page has finished printing, the page just loaded is printed while another page is loaded.

The first stage of the pipeline expands (at 763) the JPEG-compressed contone CMYK layer, expands (at 764) the Group 4 Fax-compressed bi-level black layer, and  
25 renders (at 766) the bi-level netpage tag layer according to the tag format defined in section 1.2, all in parallel. The second stage dithers (at 765) the contone CMYK layer and composites (at 765) the bi-level black layer over the resulting bi-level CMYK layer. The resultant bi-level CMYK+IR dot data is buffered and formatted (at 767) for printing on the Memjet™ printhead 350 via a set of line buffers. Most of these line buffers are  
30 stored in the off-chip DRAM. The final stage prints the six channels of bi-level dot data

(including fixative) to the Memjet™ printhead 350 via the printhead interface 768.

When several print engine controllers 760 are used in unison, such as in a duplexed configuration, they are synchronized via a shared line sync signal 770. Only one print engine 760, selected via the external master/slave pin 771, generates the line sync signal 770 onto the shared line.

The print engine controller 760 contains a low-speed processor 772 for synchronizing the page expansion and rendering pipeline, configuring the printhead 350 via a low-speed serial bus 773, and controlling the stepper motors 675, 676.

In the 8½" versions of the netpage printer, the two print engines each prints 30 Letter pages per minute along the long dimension of the page (11"), giving a line rate of 8.8 kHz at 1600 dpi. In the 12" versions of the netpage printer, the two print engines each prints 45 Letter pages per minute along the short dimension of the page (8½"), giving a line rate of 10.2 kHz. These line rates are well within the operating frequency of the Memjet™ printhead, which in the current design exceeds 30 kHz.

## 15 CONCLUSION

The present invention has been described with reference to a preferred embodiment and number of specific alternative embodiments. However, it will be appreciated by those skilled in the relevant fields that a number of other embodiments, differing from those specifically described, will also fall within the spirit and scope of the present invention. Accordingly, it will be understood that the invention is not intended to be limited to the specific embodiments described in the present specification, including documents incorporated by cross-reference as appropriate. The scope of the invention is only limited by the attached claims.